

PRINTER CONTROLLER WITH ERROR RECOVERY FOR
MULTIPLE LANGUAGE CAPABILITY

BACKGROUND OF THE INVENTION

5 The present invention relates to a print engine that includes error recovery, and in particular to a print engine with multiple language capability that includes error recovery.

10 Personal computers have become commonplace on the desks of most office workers. Typically, much of the work product of such computers is intended to be transformed into hard copy using a printer having digital imaging technology. A typical printer configuration for this purpose is a printer dedicated to each personal
15 computer. However, dedicated printers are typically inexpensive laser printers which have limited functions and features, such as a limited tray capacity and no finishing capability. More importantly, inexpensive laser printers typically handle only one page description
20 language.

25 Expensive high speed laser printers normally have flexible finishing and copy sheet capability which allows for example, custom printing and finishing of work product. Such a laser printer would be highly desirable for each personal computer, but is cost prohibitive. In
30 practice, personal computers are frequently networked together which permits a single printer to be used simultaneously by a plurality of different clients, such as personal computers or the like.

35 Referring to FIG. 1, data processing terminals 10a, 10b and 10c, such as personal computers, transmit electronic print documents through a local area network (LAN) 12 to a selected printer 14a or 14b for printing. In particular, the LAN 12 provides a medium by which
40 different terminals are able to share resources such as printers 14a, 14b, file servers, a printer server(s) 16, and scanners. Integration of shared resources has been

09128580-080498

addressed by LAN managers using different network protocols, such as Ethernet and Token Ring, to make different devices running different network protocols transparent to the terminals. Most printer servers 16 and many
5 printers 14a, 14b support a variety of print drivers using different page description languages (PDLs).

A PDL is a method of describing a printed page(s) in a printer independent format. No single standard PDL presently exists, and as a result a number
10 of industry standards have emerged. A PDL establishes an interface between a print driver or client (terminal or personal computer), and a printer server or printer. Several currently existing PDLs include PostScript (PS), Hewlett Packard Printer Control Language (HP-PCL), and
15 Interpress Page Description Language.

In existing systems, the processing of the PDL data takes place using an interpreter, which reads the received PDL data (typically text) and creates corresponding bit mapped data which is provided to the printer
20 engine. The printer engine transforms received digital data into areas of black or colored ink on the paper or other medium fed into the printer.

In networked systems, in which several users at various terminals are sharing a single printer, a desirable feature is that each terminal (i.e., each user) will
25 "see" the network accessible printer as "dedicated" to that user. In other words, when a user transmits a document to be printed on the central printer, the user desires the same service as if the printer was dedicated to that terminal. In fact, most printer interfaces used
30 on personal computers are designed to operate in a fashion that appears to the user as a one-to-one relationship between the terminal and the printer. A user does not want to observe any delays caused by an undesirable interaction of his print job with another job from a
35 different user to be printed on the central networked printer at generally the same time. Further, an

09123530 "030493

important customer requirement is that the networked printer is capable of receiving document data in a variety of different PDLs. As mentioned above, two common types of PDLs are PostScript and HP-PCL.

5 Many printers (or print servers) incorporate a plurality of language interpreters to facilitate printing documents that are transmitted to the printer in a variety of PDL formats. In such a printing system, one of several PDL interpreters within the printer (or print
10 server) is automatically selected depending upon the PDL format of the print data from a particular data processing device. Accordingly, the appropriate language interpreter must be selected for each print job. There are numerous patents describing printer controllers which
15 include multiple interpreters, such as, U.S. Patent No. 4,651,278; U.S. Patent No. 4,965,771; U.S. Patent No. 4,968,159; U.S. Patent No. 5,036,476; U.S. Patent No. 5,165,014; U.S. Patent No. 5,226,112; U.S. Patent No. 5,228,118; U.S. Patent No. 5,303,336; U.S. Patent
20 No. 5,402,527; U.S. Patent No. 5,475,801; and U.S. Patent No. 5,592,683, all of which are incorporated herein by reference.

In an ideal situation the transmission and processing of printer data is error-free. As such, all of the print jobs will likely print properly. Unfortunately, errors periodically occur in the data transmitted from the terminal to the printer, data is lost or corrupted within the printer itself, protocol failures mix different data streams at the printer server or in the printer, and errors originate within the client or printer software. For example, if PostScript data is being transmitted to the printer and an error occurs, then the printer may interpret the data as standard text which likely results in printing an extensive number of pages containing "garbled" data.

Printer description languages typically include synchronization and setup data at the beginning, and at

intermediate locations, within each document. When errors occur within a document, either a portion of the page will be corrupted while the remainder of the document prints properly, or the printer will become sufficiently confused and continue printing at the next synchronization point. Unfortunately, the next synchronization point for many documents is not until the next document, which results in the loss of the remainder of the current document. This technique of error recovery is typically used in printers that support only one PDL.

For printers that support multiple language interpreters it has been observed that frequently when an error occurs in a first document, the remainder of the document is lost in a manner similar to printers with a single language interpreter. Unfortunately, the next document is frequently likewise lost resulting in additional frustration to users.

Alternatively, the printer may request the retransmission of all or part of the PDL stream in which an error is found. However, retransmission requires a two-way communication between the printer and the originating producer of the PDL data which is typically not available in heterogeneous distributed networking environments.

What is desired, therefore, is an error recovery system for a print engine that does not result in the loss of the next document following an error.

SUMMARY OF THE PRESENT INVENTION

The present invention overcomes the aforementioned drawbacks of the prior art by providing a system for error recovery when printing documents having multiple printer description language formats. In particular, a document to be printed on a printer is received. The document is then examined to select one of a plurality of parsers suitable to parse the printer description language of the document. In response to

00128580 080498

examining the document to determine its format, the document is processed by selecting at least one of a first printer description language and a second printer description language for parsing the document. The remainder of the document is examined for synchronization data while the document is parsed.

By examining the document for synchronization data while parsing the document for printing the improved printing system does not result in the loss of subsequent documents when the synchronization data is inadvertently processed as data of a different printer description language. Preferably the system uses pointers and a buffer to permit the resetting of the page description language parsing process to the appropriate data stored within the buffer upon the detection of an error, such as when the end of document control codes in the document includes an error or is missing. For multi-stage printers that include multiple documents in the printing que which require printing, collating, and stapling, the avoidance of any preventable errors is extremely important. In the event of an unrecoverable error, it is time consuming and expensive to restart the job and clear the printer, collator, and stapler of the improperly processed job.

The foregoing and other objectives, features, and advantages of the invention will be more readily understood upon consideration of the following detailed description of the invention, taken in conjunction with the accompanying drawings.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

FIG. 1 is block diagram of a printing system including data processing terminals, local area network, printer server, and printers.

FIG. 2 is an illustration of document flow within a printer.

FIG. 3 is a block diagram of the temporal processes within current printer systems.

FIG. 4 is an illustration of document flow within a printer including an error therein.

5 FIG. 5 is a block diagram of an exemplary embodiment of the temporal processes within a printer system of the present invention.

FIG. 6 is another illustration of document flow within a printer including an error therein.

10 FIG. 7 is an illustration of the process sequence of document flow within a printer including a page description language determination, a page description language parser, pointers, and data buffer contents.

15

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

Referring to FIG. 2, printers may incorporate multiple language interpreters i, j, and k, multiple print jobs n, n+1, and n+2, each of which may have a different PDL i, j, and k, that are concatenated on a communication channel by virtue of the printer server or directly under the control of intercommunication between the printer and the individual data processing terminals 10a-10c. Print jobs are generated asynchronously, and thus may arrive at the printer in any temporal relationship. Current printers generally incorporate a mechanism for arbitrating among network clients, and a given printer will communicate with one and only one data processing terminal or server until the entirety of a print job has been received and placed into an internal queue (or processed in its entirety) before engaging in negotiation with another terminal or server to receive a subsequent print job. Many printers contain sufficient memory and/or storage to hold multiple pending print jobs while a current job is being interpreted and printed. Therefore, it is clear that the communication channel between the printer and the originator of a print job

30

35

09123580 "080498

data stream may have been terminated before the contents of the data stream are parsed and processed.

Referring to FIG. 3, typical processing elements within a printer include a series of processes and steps that are executed in a particular predefined manner. When there are no prior documents in the print queue of the printer, a new document data stream is received by the master control process 40 which manages the hardware of the printer and the different document flows through the printer. As such, the master control process 40 receives and responds to internal and external data sensor input, as well as managing the initiation and termination of other subprocesses within the printer. After receipt of a document, the master control process 40 initiates a data receiver process 42 which manages communication within the local area network, reads the data stream from a print job, and stores data in an internal data buffer. Next, a printer description language (PDL) determination process 44 examines the current contents of the data buffer, and determines the particular PDL of the initial portion of the data buffer contents. This process permits the identification of the particular PDL language encoding of the document. The data receiver process 42 may terminate prior to the completion of the PDL determination process 44 or may continue concurrently with the process, depending on the characteristics of the printer and the state of network print job requests. Once the determination is made as to which PDL language the document is encoded in, the PDL determination process 44 is terminated because the remainder of the document is always encoded using the same technique. Then control is passed to the PDL parsing process 46, which reads data from the buffer and creates the corresponding output bitmap which is then transformed into the printed page(s).

With such a printing process, it has been accepted that if an error occurs that the PDL parsing

09128580 080498

process 46 is unable to recover from, then the PDL parsing process 46 will terminate and notify the master control process 40 of the error. The master control process 40 then starts both the data receiver process 42, 5 if deactivated, and starts the PDL determination process 44 to search for the next synchronization point. The next synchronization point in many cases is the following document. Also, several pages of "garbage" will likely print upon the occurrence of an error. Moreover, the 10 next document is frequently lost resulting in additional user frustration.

Referring to FIG. 4, the present inventors came to the realization that the principal reason that the printing process periodically discards the second document 62 after an error occurs in the first document 60 is 15 that the end of document control codes 64 in the first document include an error or are missing. Without the end of document control codes 64, the PDL parsing process 46 attempts to parse the synchronization data 66 of the 20 second document 62 as if it were data of the previous document. In such an event, several lines of synchronization data 66 of the second document 62 are processed before the PDL parsing process 46 determines that an error has occurred. The PDL parsing process 46 then 25 notifies the master control process 40 of the error which then reinitializes the data receiver process 42 and the PDL determination process 44. The PDL determination process then looks for the next synchronization point. Normally, this results in waiting until a third document 30 70 because the synchronization data 66 of the second document 62 has already been inappropriately processed as data by the PDL parsing process 46. Periodically, while searching for synchronization data, the PDL determination process 44 locates data within the second document 62 35 that matches another printer description language. The PDL parsing process 46 will then attempt to process the remaining second document 62 with the matched printer

09128580 "080498

description language, which may not be the proper printer description language for the second document 62, resulting in many pages of "garbage."

With the aforementioned realization of the reason why the printing process periodically discards the second document 62 after an error occurs in the first document 60, the present inventors developed an improved printing system to overcome this limitation. Referring to FIG. 5, the master control process 80 initiates the data receiver process 82, and when sufficient data is present in the data stream buffer 83, the PDL determination process 84 is initiated to determine the PDL of the data stream. However, once the PDL is identified and the appropriate PDL parsing process 86, 88, 90, or 92 is initiated, the PDL determination process 84 continues to execute, reading characters from the data stream buffer 83 and determining potential points at which the PDL may have changed. As segments of the data stream are processed (for example, at the end of each page), the selected PDL parsing process 86, 88, 90 or 92 sends a signal to the PDL determination process 84 to advance its pointer in a data buffer to the most recent data byte which has been parsed with no ambiguity.

Referring to FIG. 6, if three data streams n 100, $n+1$ 102 and $n+2$ 104 are received in sequence by the data receiver process 82, prior to the completion of processing of the first data stream 100, the data streams are stored in the data buffer. To illustrate the improved technique, assume that an error exists in data stream $n+1$ 102. As data stream n 100 is processed, the PDL determination process 84 recognizes that data stream n 100 as consisting of PDL i commands 106 and accordingly the PDL i parsing process 86 is activated. When job n 100 is finished without error, the PDL determination process 84 is reinitialized and will proceed to analyze data stream $n+1$ 102 and recognizing it as consisting of PDL j commands 108. The PDL j parsing process 88 will be

09128580 080498

activated. As segments of job n+1 102 are completed, the data buffer corresponding to the PDL determination process 84 is advanced in stages, and the PDL determination process 84 continues to evaluate the buffer contents to determine the PDL language of the contents starting at each intermediate point. When erroneous data is detected 112 in data stream n+2 104 (such as missing or corrupted end of document control codes 110), the PDL parsing process 88 exits with an error condition and the PDL determination process 84 immediately determines the most probable language corresponding to the buffer contents at the last prior intermediate point signaled by the PDL parsing process 88. In this example, the PDL determination process 84 requires the presence of an appropriate preamble or header for a valid data stream prior to signaling recognition of the PDL, after which the corresponding PDL parsing process 86, 88, 90, and 92 is activated. A buffer pointer corresponding to the position in the buffer where the PDL determination process 84 found an appropriate preamble or header may be prior to the position at which the previous PDL parsing process 88 found the error. In this example, the buffer pointer is reset to the start of data stream 104 and the PDL determination process 84 actuates PDL parsing process 90 for PDL k. The result is proper printing of data stream n+2 104.

Referring to FIG. 7, the temporal sequence of operations illustrates that the PDL parsing operation stays active, while keeping track of a position in the data buffer at which the identify of the PDL of the corresponding data was relatively certain. When an error is encountered, the PDL determination process can reestablish the identify of the PDL stream in the buffer starting from the prior acknowledgment point, and prevent the loss of a complete job from the data stream. Conventional processing for the documents of FIG. 7 would have resulted in the loss of document n+2. Under the improved

09128580 080498

processing document n+2 is not lost because the error is detected and the parser is reset to the beginning of document n+2 from the buffer.

Accordingly, the improved printing process does not result in the loss of subsequent documents when the synchronization data is inadvertently processed as data of a different printer description language. The use of pointers and the buffer permits the resetting of the PDL parsing process to the appropriate data stored within the buffer upon the detection of an error, such as when the end of document control codes in the document includes an error or is missing. For multi-stage printers that include multiple documents in the printing que which require printing, collating, and stapling, the avoidance of any preventable errors is extremely important. In the event of an unrecoverable error, it is time consuming and expensive to restart the job and clear the printer, collator, and stapler of the improperly processed job.

The terms and expressions which have been employed in the foregoing specification are used therein as terms of description and not of limitation, and there is no intention, in the use of such terms and expressions, of excluding equivalents of the features shown and described or portions thereof, it being recognized that the scope of the invention is defined and limited only by the claims which follow.

09123580 080498